



**Universidade Federal do Ceará**  
**Curso de Tecnologia em Ciência de Dados**  
**Redes Neurais Artificiais**  
**Atividade teórica – Lista I – Respostas**

**Aluno (a):** Larissa Vitória Vasconcelos Sousa

**Turma:** CD5

**Professor:** Hitalo Joseferson Batista Nascimento

**Matrícula:** 519221

**Data:** 02/10/2023

**a) Exercícios - revisão de IA**

**Capítulo 1 (1.1, 1.7, 1.10, 1.11, 1.12, 1.14)**

**1.1 a) Inteligência:** é a capacidade de uma pessoa ou um sistema artificial, de aprender, compreender, adaptar-se ao ambiente, resolver problemas, tomar decisões informadas e executar tarefas de forma eficaz.

**b) Inteligência Artificial:** conhecida também como IA, ela é um campo da informática que se concentra no desenvolvimento de sistemas e programas de computador que podem realizar tarefas que normalmente exigiriam inteligência humana. Isso inclui o uso de algoritmos, aprendizado de máquina e processamento de dados para automatizar tarefas, tomar decisões e resolver problemas complexos.

**c) Agente:** é uma entidade, seja humana, animal, ou um programa de computador, que age e interage com seu ambiente para atingir objetivos específicos. Os agentes podem ser simples, como um termostato que regula a temperatura em uma sala, ou complexos, como um robô autônomo que navega em um ambiente desconhecido.

**d) Racionalidade:** se refere à capacidade de tomar decisões ou realizar ações que são consistentes com os objetivos estabelecidos e que maximizam a probabilidade de alcançar esses objetivos.

**e) Raciocínio Lógico:** é um processo mental no qual se aplicam regras formais de inferência para chegar a conclusões baseadas em premissas ou evidências.

**1.7 • Leitores de código de barra de supermercados.**

Os leitores de código de barras de supermercados são dispositivos simples que escaneiam códigos de barra e recuperam informações associadas a produtos, como preço e descrição. Eles não incorporam um alto grau de inteligência artificial, porque operam com base em código de barras lido e as informações armazenadas em um

banco de dados. Essa é uma aplicação de automação de tarefas específica, mas não envolve aprendizado ou adaptação significativa.

- **Menus de voz de telefones.**

Os menus de voz de telefones incorporam elementos de inteligência artificial, especialmente quando usam reconhecimento de fala e processamento de linguagem natural para entender e responder às solicitações dos usuários.

- **Mecanismos de busca na Web.**

Os mecanismos de busca na Web usam inteligência artificial. Eles usam algoritmos complexos de aprendizado de máquina e processamento de linguagem natural para indexar e classificar bilhões de páginas da web, entender as intenções dos usuários e fornecer resultados relevantes.

- **Algoritmos de roteamento da Internet que respondem dinamicamente ao estado da rede.**

Os algoritmos de roteamento da Internet são sistemas de automação avançados e altamente especializados que gerenciam o encaminhamento de dados na Internet. Eles podem incorporar inteligência artificial, como a capacidade de tomar decisões com base em métricas de desempenho e condições de rede em tempo real.

**1.10** A Inteligência Artificial (IA) é uma mistura de ciência e engenharia. Na ciência, os pesquisadores estudam e desenvolvem teorias e algoritmos relacionados à IA, como aprendizado de máquina e processamento de linguagem natural. Já na engenharia, os engenheiros de IA aplicam essas teorias para criar sistemas práticos, como assistentes virtuais e carros autônomos.

**1.11** A afirmação é verdadeira. Os computadores não têm autonomia ou inteligência própria, e sua funcionalidade é determinada pelos programas e algoritmos criados por programadores humanos.

**1.12** Mesmo os genes desempenhando um papel na determinação das características e comportamentos dos animais, a inteligência animal é influenciada por uma combinação de fatores genéticos e ambientais. Os animais podem aprender com base em experiências e interações com o ambiente.

**1.14 a) Jogar um jogo decente de tênis de mesa (pingue-pongue).**

O tênis de mesa é um esporte que envolve uma interação física complexa. Embora existam robôs que podem até jogar tênis de mesa, mas ainda é um desafio para a IA competir em níveis humanos de habilidade devido à necessidade de interações físicas precisas.

**b) Dirigir no centro do Cairo, Egito.**

A direção autônoma em áreas de tráfego intenso e complexo, como o centro do Cairo, é um desafio. A IA tem feito avanços em direção autônoma, mas ainda não atingiu um nível onde possa lidar facilmente com todas as situações do trânsito em cidades caóticas como o Cairo.

**c) Dirigir em Victorville, Califórnia.**

Em áreas menos congestionadas e mais estruturadas, a direção autônoma é mais viável. Empresas de tecnologia estão testando veículos autônomos em áreas como Victorville, mas a tecnologia ainda está em desenvolvimento.

**d) Comprar mantimentos para uma semana no mercado.**

Os assistentes de compras baseados em IA podem ajudar a fazer compras, em alguns lugares já são assim, há somente o caixa mas sem funcionário.

**e) Comprar uma semana de mantimentos na Web.**

A compra de mantimentos on-line com ajuda de IA é comum e está em crescimento. Algoritmos de recomendação podem sugerir produtos com base no histórico de compras do usuário.

**f) Jogar um jogo decente de bridge em nível competitivo.**

É um desafio para a IA, devido à complexidade das regras e estratégias envolvidas. No entanto, sistemas de IA têm feito avanços nessa área e podem competir em níveis respeitáveis, embora ainda haja espaço para melhorias.

**g) Descobrir e provar novos teoremas matemáticos.**

A IA tem sido usada com sucesso em matemática para auxiliar na descoberta de novos teoremas e na prova de conjecturas. No entanto, a criatividade matemática humana ainda desempenha um papel fundamental nesse processo.

**h) Escrever uma história intencionalmente engraçada.**

A geração de conteúdo criativo e engraçado usando IA é possível, tendo em vista as muitas IA's que estão aí, como por exemplo o ChatGPT.

**i) Dar assessoria jurídica competente em uma área especializada de direito.**

A IA tem sido usada em pesquisa jurídica e análise de documentos, mas fornecer assessoria jurídica competente em áreas especializadas envolve interpretação legal complexa, que ainda é uma tarefa humana.

**j) Traduzir inglês falado em sueco falado, em tempo real.**

A tradução de fala em tempo real é uma área ativa de pesquisa em IA, e os sistemas de tradução têm melhorado significativamente. No entanto, traduzir fala com precisão em tempo real ainda é um desafio.

**k) Executar uma operação cirúrgica complexa.**

A cirurgia robótica assistida por IA está em uso, mas geralmente é supervisionada por cirurgiões humanos. A IA pode auxiliar em procedimentos cirúrgicos, mas a tomada de decisões críticas ainda é realizada por profissionais de saúde.

## **Capítulo 2 (2.1 ao 2.13)**

**2.1** Quando a medição de desempenho considera apenas os primeiros  $T$  passos de tempo, as ações do agente racional dependem do estado do ambiente e do passo de tempo atual, pois o agente deve levar em consideração o tempo restante disponível ( $T$  - sem tempo atual) para otimizar seu comportamento e alcançar seus objetivos dentro deste prazo.

**2.2 a) Mostre que a função do agente aspirador de pó simples descrito na Figura 2.3 é realmente racional, conforme as suposições listadas na página 38.**

A função do agente de vácuo simples da Figura 2.3 é razoável desde que satisfaça as suposições da página 38, como perceber com precisão seu estado, ser capaz de influenciar o ambiente e buscar ações que maximizem a medida do desempenho (sujeira limpa).

**b) Descreva uma função de agente racional para o caso em que cada movimento custa um ponto. O programa de agente correspondente exige estado interno?**

No caso em que cada movimento custa um ponto, uma função racional do agente pode ser minimizar o número de movimentos para limpar todos os quadrados sujos. Um estado interno não é necessário para esta ação, uma vez que as ações podem ser determinadas por observações imediatas.

**c) Descreva possíveis projetos de agentes para os casos em que quadrados limpos podem ficar sujos e a geografia do ambiente é desconhecida. Faz sentido para o agente aprender a partir de sua experiência nessas situações? Em caso afirmativo, o que ele deve aprender? Se não, por quê?**

Em ambientes onde quadrados limpos podem ficar sujos e a geografia é desconhecida, faz sentido que um agente aprenda com a experiência. O agente deve aprender a reconhecer padrões de sujeira e ajustar seu comportamento para otimizar a limpeza. O que precisa de aprender depende do ambiente específico, mas pode incluir estratégias de pesquisa e respostas eficazes às mudanças ambientais. O espaço interior pode ser útil para encontrar informações relevantes, mas não é absolutamente necessário.

**2.3 a) Um agente que detecta apenas informações parciais sobre o estado não pode ser perfeitamente racional.**

Verdadeiro. Um agente que detecta apenas informações parciais sobre o estado do ambiente não pode ser perfeitamente racional, pois ele pode tomar decisões subótimas devido à falta de informações completas. Por exemplo, se um robô aspirador de pó não consegue detectar sujeira em uma parte do ambiente devido a sensores defeituosos, ele pode não ser capaz de manter o ambiente completamente limpo, mesmo que seja programado para fazê-lo. A falta de informações completas pode levar a ações imperfeitas.

**b) Existem ambientes de tarefa nos quais nenhum agente reativo puro pode comportar-se racionalmente.**

Verdadeiro. Existem ambientes de tarefa nos quais nenhum agente reativo puro (que toma decisões apenas com base no estado atual) pode se comportar racionalmente, pois a racionalidade pode exigir considerar informações passadas ou futuras.

**c) Existe um ambiente de tarefa em que todo agente é racional.**

Falso. Não existe um ambiente de tarefa em que todo agente é racional, pois a racionalidade depende das metas e medidas de desempenho específicas do agente em relação ao ambiente.

**d) A entrada para o programa de agente é a mesma que a entrada para a função de agente.**

Verdadeiro. A entrada para o programa de agente (ou seja, a função de mapeamento de percepções para ações) é a mesma que a entrada para a função de agente (ou seja, o mapeamento de estados para ações).

**e) Toda função de agente é implementável por uma combinação de programa/máquina.**

Verdadeiro. Toda função de agente é implementável por uma combinação adequada de programa e máquina que realiza essa função.

**f) Suponha que um agente selecione sua ação uniformemente ao acaso do conjunto de ações possíveis. Existe um ambiente de tarefa determinista em que esse agente é racional.**

Verdadeiro. Existe um ambiente de tarefa determinista em que um agente que seleciona ações uniformemente ao acaso não é racional, pois ele não estaria otimizando o desempenho de acordo com a medida de desempenho.

**g) É possível para um dado agente ser perfeitamente racional em dois ambientes de tarefa distintos.**

Verdadeiro. Um agente pode ser perfeitamente racional em dois ambientes de tarefa distintos, desde que ele se adapte e aja de maneira otimizada de acordo com as características específicas de cada ambiente.

**h) Todo agente é racional em um ambiente não observável.**

Falso. Nem todo agente é racional em um ambiente não observável, pois a racionalidade muitas vezes depende da informação disponível para tomar decisões adequadas.

**i) Um agente jogador de pôquer perfeitamente racional nunca perde.**

Falso. Um agente jogador de pôquer perfeitamente racional pode perder jogos devido à natureza do jogo, à aleatoriedade das cartas ou às ações dos outros jogadores, mesmo que tome as melhores decisões possíveis com base nas informações disponíveis. A racionalidade não garante vitórias consistentes em jogos de azar ou estratégia onde a incerteza está presente.

## **2.4 • Jogar futebol.**

Ambiente da Tarefa (PEAS): Campo de futebol, bola, jogadores, árbitro. Propriedades: Totalmente observável, determinístico (exceto decisões arbitrárias do árbitro), sequencial (ações em sequência), estático (exceto movimento da bola e jogadores), multiagente (mais de um jogador interagindo), competitivo (objetivo oposto), desconhecido (intenções dos oponentes).

### **• Explorar os oceanos subterrâneos de Titã.**

Ambiente da Tarefa (PEAS): Subsuperfície de Titã, veículo de exploração, sensores. Propriedades: Parcialmente observável (limitações dos sensores), estocástico (desconhecimento da topologia), sequencial (exploração em etapas), dinâmico (movimento do veículo), desconhecido (descobertas científicas).

### **• Comprar livros usados de IA na Internet.**

Ambiente da Tarefa (PEAS): Site de compra online, catálogo de livros, vendedores, comprador. Propriedades: Totalmente observável, determinístico (na maioria dos casos), episódico (decisões independentes), estático (exceto atualização do catálogo), multiagente (vendedores independentes), colaborativo (negociação).

- **Jogar uma partida de tênis.**

Ambiente da Tarefa (PEAS): Quadra de tênis, raquetes, bola, jogadores. Propriedades: Totalmente observável, determinístico (exceto eventos imprevisíveis), sequencial (troca de saques), dinâmico (movimento da bola e jogadores), multiagente (dois jogadores), competitivo (objetivo oposto).

- **Praticar tênis contra uma parede.**

Ambiente da Tarefa (PEAS): Parede, raquete, bola. Propriedades: Totalmente observável, determinístico (exceto erros do jogador), sequencial (troca de golpes), estático (parede imóvel), singular (apenas um jogador).

- **Realizar um salto de altura.**

Ambiente da Tarefa (PEAS): Pista de salto de altura, vara, atleta, obstáculo. Propriedades: Totalmente observável, determinístico (condições da pista), sequencial (preparação e execução), dinâmico (movimento do atleta), singular (um atleta).

- **Licitações de um item em um leilão.**

Ambiente da Tarefa (PEAS): Leilão, item, licitantes, leiloeiro. Propriedades: Totalmente observável, estocástico (lances dos concorrentes), sequencial (licitações em rodadas), dinâmico (decisões dos licitantes), multiagente (licitantes independentes), competitivo (disputa pelo item), colaborativo (estratégias de licitação).

**2.5 Agente:** Um agente é uma entidade capaz de interagir com um ambiente para atingir seus objetivos. Pode ser um programa de computador, um robô físico ou até mesmo um ser humano.

**Função de Agente:** A função de agente descreve a relação entre as percepções do agente (informações do ambiente) e as ações que o agente toma para maximizar sua medida de desempenho (atingir seus objetivos).

**Programa de Agente:** O programa de agente é a implementação concreta da função de agente. É o código ou o conjunto de regras que o agente segue para tomar decisões com base em suas percepções.

**Racionalidade:** A racionalidade de um agente refere-se à capacidade do agente de tomar as melhores ações possíveis com base em suas informações disponíveis e objetivos. Um agente racional busca maximizar sua medida de desempenho.

**Autonomia:** A autonomia de um agente se refere à sua capacidade de agir independentemente, tomar decisões e executar ações sem intervenção externa. Um agente autônomo é capaz de agir por conta própria.

**Agente Reativo:** Um agente reativo toma decisões com base apenas nas percepções imediatas do ambiente, sem manter um estado interno ou modelo do mundo. Suas ações são determinadas por regras diretas que mapeiam percepções para ações.

**Agente Baseado em Modelo:** Um agente baseado em modelo mantém uma representação interna do ambiente (um modelo) e toma decisões com base nesse modelo. Isso permite que o agente faça previsões sobre o futuro e planeje suas ações de acordo.

**Agente Baseado em Objetivos:** Um agente baseado em objetivos define explicitamente seus objetivos e busca ações que o aproximem desses objetivos. Ele pode considerar diferentes ações e seus efeitos no estado do mundo.

**Agente Baseado em Utilidade:** Um agente baseado em utilidade leva em consideração não apenas a realização dos objetivos, mas também a utilidade associada a diferentes resultados. Ele busca maximizar a utilidade esperada, considerando incertezas no ambiente.

**Agente com Aprendizagem:** Um agente com aprendizagem é capaz de melhorar seu desempenho ao longo do tempo por meio da experiência. Ele utiliza técnicas de aprendizado de máquina para adaptar suas ações com base em dados passados e novas informações do ambiente.

**2.6 a) Pode haver mais de um programa de agente que implemente uma dada função de agente? Dê um exemplo ou mostre por que não é possível.**

Sim, pode haver mais de um programa de agente que implemente a mesma função de agente. Por exemplo, para uma função de agente que mapeia percepções simples para ações, diferentes programas de agente podem ser escritos para alcançar o mesmo mapeamento de entrada e saída.

**b) Existem funções de agente que não podem ser implementadas por qualquer programa de agente?**

Sim, existem funções de agente que não podem ser implementadas por nenhum programa de agente. Isso ocorre quando a função requer capacidades computacionais específicas que não estão disponíveis em uma máquina ou programa.

**c) Dada uma arquitetura de máquina fixa, cada programa de agente implementa exatamente uma função de agente?**

Não, nem sempre. Dada uma arquitetura de máquina fixa, pode haver programas de agente que implementem funções de agente diferentes, dependendo das instruções e da lógica de processamento específicas do programa.

**d) Dada uma arquitetura com  $n$  bits de armazenamento, quantos programas de agentes distintos são possíveis nessa arquitetura?**

O número de programas de agentes distintos possíveis em uma arquitetura com  $n$  bits de armazenamento é  $2^n$ , pois cada bit pode ser configurado de duas maneiras (0 ou 1), e há  $n$  bits disponíveis.

**e) Suponha manter fixo o programa de agente, mas aumentamos a velocidade da máquina por um fator de dois. Isso muda a função de agente?**

Aumentar a velocidade da máquina por um fator de dois não altera a função de agente, desde que a lógica do programa de agente permaneça inalterada. A função de agente continua a ser a mesma, apenas as ações são executadas mais

rapidamente.

**2.7, 2.8, 2.9** - códigos em outro arquivo.

**2.10 a) Um agente reativo simples pode ser perfeitamente racional para esse ambiente? Explique.**

Um agente reativo simples não pode ser perfeitamente racional para esse ambiente modificado, no qual o agente é penalizado por cada movimento. Isso ocorre porque o agente reativo simples não leva em consideração as penalidades associadas aos movimentos. Ele continuaria se movendo aleatoriamente ou em uma direção fixa, sem considerar as penalidades, o que resultaria em um desempenho subótimo.

**b) E um agente reativo com estado? Projete tal agente.**

Um agente reativo com estado pode ser projetado para lidar melhor com o ambiente modificado, levando em consideração as penalidades por movimento. O agente pode manter um estado interno que inclui informações sobre sua posição atual, bem como o estado de limpeza ou sujeira de cada quadrado no ambiente. Ele pode, então, tomar decisões com base em minimizar o número total de movimentos, levando em conta as penalidades.

Código em outro arquivo da pasta.

**c) Como suas respostas para os itens a e b mudarão se as percepções do agente fornecerem o status limpo/sujo de cada quadrado no ambiente?**

**2.11 a) Um agente reativo simples pode ser perfeitamente racional para esse ambiente? Explique.**

Um agente reativo simples não pode ser perfeitamente racional para um ambiente com geografia desconhecida, onde os limites e obstáculos não são conhecidos. Isso ocorre porque um agente reativo simples toma decisões com base apenas em percepções locais e não possui informações sobre a geografia completa do ambiente. Portanto, ele pode ficar preso em obstáculos ou não conseguir navegar eficazmente em ambientes desconhecidos.

**b) Um agente reativo simples com uma função de agente aleatório pode superar um agente reativo simples? Projete tal agente e faça a medição de seu desempenho em vários ambientes.**

Um agente reativo simples com uma função de agente aleatória não necessariamente superará um agente reativo simples. Uma função de agente aleatória tomará ações de forma totalmente aleatória, o que significa que o agente pode tomar decisões ineficazes e incoerentes. Em alguns casos, o agente aleatório pode acidentalmente tomar ações que o levam a resultados melhores do que o agente reativo simples, mas isso seria puramente aleatório e não refletiria uma melhoria consistente de desempenho.

Código em outro arquivo da pasta.

**c) Você poderia projetar um ambiente no qual seu agente aleatório tenha um desempenho muito ruim? Mostre seus resultados.**



Sim, é possível projetar um ambiente no qual um agente aleatório tenha um desempenho muito ruim. Isso pode ser alcançado criando um ambiente com muitos obstáculos e sujeira, onde as ações corretas são necessárias para evitar obstáculos e limpar a sujeira. O agente aleatório teria uma alta probabilidade de tomar ações que o levam a ficar preso em obstáculos ou não realizar a limpeza efetiva.

**d) Um agente reativo com estado pode superar um agente reativo simples? Projete tal agente e faça a medição de seu desempenho em vários ambientes. Você pode projetar um agente racional desse tipo?**

Um agente reativo com estado pode superar um agente reativo simples em um ambiente com geografia desconhecida. O agente com estado pode manter uma representação interna do ambiente à medida que explora e aprender com suas percepções anteriores. Isso permite que o agente tome decisões mais informadas e evite obstáculos, mesmo em um ambiente desconhecido. No entanto, a perfeita racionalidade dependeria da capacidade do agente em inferir a geografia do ambiente com base em suas percepções, o que pode não ser possível em todos os casos. Portanto, a perfeita racionalidade pode não ser alcançada em todos os ambientes desconhecidos.

**2.12** Um agente reativo simples não pode ser perfeitamente racional em um ambiente com geografia desconhecida e um sensor de impacto. Ele ainda não possui informações sobre a geografia do ambiente, e o sensor de impacto apenas detecta tentativas de movimento, não fornecendo informações sobre a localização real dos obstáculos. Um agente reativo simples com uma função de agente aleatório pode superar um agente reativo simples apenas em alguns casos aleatórios. O sensor de impacto não fornece informações suficientes para o agente tomar decisões informadas sobre como evitar obstáculos ou cruzar limites. Se o sensor de impacto parar de funcionar, o agente não terá conhecimento de obstáculos ou limites e continuará tentando se mover para locais onde não deveria. Isso pode resultar em comportamento ineficaz e possivelmente danos ao agente ou ao ambiente. O agente precisaria de informações adicionais ou uma nova maneira de perceber os obstáculos e limites para se comportar de maneira racional

**2.13 a) Um agente reativo simples pode ser perfeitamente racional para esse ambiente? Explique.**

Lei de Murphy (25% de falha ao Aspirar e sensor de sujeira incorreto 10% do tempo): Nesse ambiente estocástico, onde as ações do agente e os sensores podem falhar, o programa de agente precisa ser mais robusto. Um agente racional pode ser projetado da seguinte forma:

Código em outro arquivo da pasta.

**b) Um agente reativo simples com uma função de agente aleatório pode superar um agente reativo simples? Projete tal agente e faça a medição de seu desempenho em vários ambientes.**

Crianças Pequenas (10% de chance de um quadrado limpo ficar sujo a cada período de tempo): Neste ambiente estocástico, onde a sujeira aparece

aleatoriamente, um agente racional pode ser projetado para priorizar a limpeza. Aqui está um exemplo de um agente que tenta manter o ambiente limpo:  
Código em outro arquivo da pasta.

**c) Você poderia projetar um ambiente no qual seu agente aleatório tenha um desempenho muito ruim? Mostre seus resultados.**

É possível projetar um ambiente que desafie significativamente um agente aleatório, tornando seu desempenho muito ruim. Um exemplo seria um ambiente onde a sujeira aparece de forma extremamente imprevisível e com alta frequência. Nesse caso, um agente aleatório teria dificuldade em tomar decisões eficazes, já que não haveria um padrão claro para seguir. As chances de sucesso de um agente aleatório dependeriam quase inteiramente da sorte.

**d) Um agente reativo com estado pode superar um agente reativo simples? Projete tal agente e faça a medição de seu desempenho em vários ambientes. Você pode projetar um agente racional desse tipo?**

Um agente reativo com estado pode superar um agente reativo simples em ambientes mais complexos, nos quais o histórico de ações e percepções é relevante para tomar decisões adequadas. O agente reativo com estado mantém informações sobre o estado atual do ambiente e pode usar esse conhecimento para tomar decisões melhores. Por exemplo, se o ambiente for expansivo e houver múltiplas salas, um agente reativo com estado pode lembrar quais salas já foram limpas e quais ainda precisam ser limpas. Ele pode priorizar a limpeza das salas que ainda não foram limpas. Além disso, se o ambiente for estocástico, como no caso da Lei de Murphy, um agente com estado pode aprender com experiências passadas e adaptar seu comportamento com base nesse aprendizado.

### **Capítulo 3 (3.1, 3.2, 3.3, 3.7, 3.9, 3.14)**

**3.1** A ideia de que a formulação do problema deve seguir a formulação do objetivo é um princípio importante em inteligência artificial e resolução de problemas. Isso se relaciona com a clareza e a eficácia na definição de um problema a ser resolvido.

**3.2 a) Formule este problema. Qual é o tamanho do espaço de estados?**

Formulação do Problema:

Estado Inicial: Robô no meio do labirinto, orientado para o norte.

Ações Possíveis: Virar para norte, sul, leste ou oeste; mover-se uma certa distância para frente.

Estado Objetivo: Qualquer estado fora do labirinto.

Espaço de Estados: O tamanho do espaço de estados depende do tamanho do labirinto, mas é geralmente grande, pois inclui todas as possíveis posições e orientações do robô dentro do labirinto.

**b) Ao navegar pelo labirinto, é necessário virar apenas na interseção de dois ou mais corredores. Reformule esse problema usando essa observação. Qual será o tamanho do espaço de estados agora?**

Reformulação com Observação de Interseções:

Estado Inicial: Robô no meio do labirinto, orientado para o norte.

Ações Possíveis: Mover-se para a próxima interseção e virar.

Estado Objetivo: Qualquer estado fora do labirinto.

Espaço de Estados: O espaço de estados é menor, pois agora só consideramos interseções como estados possíveis.

**c) De qualquer ponto do labirinto, podemos mover em qualquer uma das quatro direções, até ter alcançado um ponto de virar e essa é a única ação que precisa ser feita. Reformule o problema usando essas ações. É necessário acompanhar a orientação do robô para resolver esse problema?**

Reformulação com Ações Simplificadas:

Estado Inicial: Qualquer posição dentro do labirinto.

Ações Possíveis: Mover-se em qualquer uma das quatro direções até chegar a uma interseção e, em seguida, virar.

Estado Objetivo: Qualquer estado fora do labirinto.

Orientação: A orientação do robô não é mais relevante, pois ele só precisa se movimentar e virar nas interseções.

**d) Na descrição inicial do problema já abstraímos do mundo real, restringindo as ações e removendo os detalhes. Liste três simplificações que fizemos.**

Reformulação com Ações Simplificadas:

Estado Inicial: Qualquer posição dentro do labirinto.

Ações Possíveis: Mover-se em qualquer uma das quatro direções até chegar a uma interseção e, em seguida, virar.

Estado Objetivo: Qualquer estado fora do labirinto.

Orientação: A orientação do robô não é mais relevante, pois ele só precisa se movimentar e virar nas interseções.

**3.3 a) Escreva uma formulação detalhada para esse problema de busca (será útil definir alguma notação formal).**

Formulação do Problema de Busca:

Estado: A posição atual de ambos os amigos no mapa, representada como (cidade\_A, cidade\_B).

Ações Possíveis: Mover cada amigo individualmente para uma cidade vizinha no mapa.

Custo do Caminho: A distância da estrada entre as cidades onde cada amigo se move.

Estado Objetivo: Ambos os amigos estão na mesma cidade.

Função de Sucessor: Define as ações possíveis de acordo com as cidades vizinhas no mapa.

Função de Custo: Define o custo das ações de acordo com as distâncias da estrada.

**b) Seja  $D(i, j)$  a distância em linha reta entre as cidades  $i$  e  $j$ . Qual das seguintes funções heurísticas é admissível? (i)  $D(i, j)$ , (ii)  $2 \cdot D(i, j)$ , (iii)  $D(i, j)/2$ .**

Funções Heurísticas Admissíveis:

(i)  $D(i, j)$ : Esta é uma função heurística admissível, pois fornece uma estimativa otimista (nunca superestima o custo) da distância real entre as cidades  $i$  e  $j$ .

(ii)  $2 \cdot D(i, j)$ : Esta função também é admissível, pois ainda fornece uma estimativa otimista, mas pode ser excessivamente otimista, o que pode levar a uma busca menos eficiente.

(iii)  $D(i, j)/2$ : Esta função é admissível, mas tende a subestimar o custo, o que pode levar a uma busca mais lenta.

**c) Há mapas completamente conectados para os quais não existe solução?**

Mapas Completamente Conectados:

Não deve haver mapas completamente conectados para os quais não existe solução. Em um mapa completamente conectado, é possível alcançar qualquer cidade a partir de qualquer outra cidade. Portanto, os amigos sempre podem se encontrar movendo-se para a cidade do outro.

**d) Há mapas em que todas as soluções requerem que um amigo visite a mesma cidade duas vezes?**

Mapas com Visitas Duplas:

Em mapas onde todas as estradas têm a mesma distância, todas as soluções requerem que um amigo visite a mesma cidade duas vezes. Isso ocorre porque, após um amigo alcançar a cidade do outro, ambos devem se mover para uma cidade intermediária para evitar que um amigo espere indefinidamente. Portanto, em tais mapas, a visita dupla é inevitável para otimizar o tempo de encontro.

**3.7 a) Suponha que o espaço de estados consista de todas as posições  $(x, y)$  do plano. Quantos estados existem? Quantos caminhos existem até o objetivo?**

Espaço de Estados:

O espaço de estados consiste em todas as posições  $(x, y)$  do plano, o que significa que há infinitos estados possíveis, uma vez que o plano é contínuo. Não há um número finito de estados neste caso. O número de caminhos possíveis até o objetivo também é infinito, pois existem infinitas maneiras de traçar um caminho contínuo entre dois pontos em um plano contínuo.

**b) Explique brevemente por que o caminho mais curto de um vértice de um polígono até qualquer outro vértice na cena deve consistir de segmentos de reta que unem alguns vértices dos polígonos. Agora defina um bom espaço de estados. Qual é o tamanho desse espaço de estados?**

Caminho Mais Curto em Ambiente com Obstáculos:

O caminho mais curto de um vértice de um polígono até qualquer outro vértice na cena deve consistir em segmentos de reta que unem alguns vértices dos polígonos, devido à natureza convexa dos obstáculos poligonais convexos. Isso ocorre porque, em um ambiente com obstáculos convexos, a linha reta entre dois pontos é a trajetória mais curta e eficiente.

Um bom espaço de estados poderia ser definido como um grafo de visibilidade, onde os vértices do grafo são os vértices dos polígonos e quaisquer pontos de

interseção entre segmentos de reta que conectam os vértices. O tamanho deste espaço de estados depende do número de vértices e interseções.

**c) Defina as funções necessárias para implementar o problema de busca, incluindo uma função AÇÕES que receba um vértice como entrada e devolve um conjunto de vetores, sendo que cada vetor mapeia o vértice atual a um dos vértices que pode ser alcançado por uma linha reta. (Não se esqueça dos vizinhos no mesmo polígono.) Utilize a distância em linha reta como função heurística.**

Funções Necessárias:

A função AÇÕES deve receber um vértice como entrada e devolver um conjunto de vetores que mapeiam o vértice atual para outros vértices acessíveis por segmentos de reta. Isso inclui os vizinhos no mesmo polígono e os vértices de outros polígonos que podem ser alcançados diretamente em linha reta.

A função heurística pode ser a distância em linha reta (distância euclidiana) entre um vértice e o objetivo.

**d) Aplique um ou mais algoritmos deste capítulo para resolver alguns problemas nesse domínio e comente seu desempenho.**

Desempenho dos Algoritmos:

Os algoritmos de busca, como A\* (A estrela), podem ser aplicados com sucesso para encontrar o caminho mais curto em ambientes com obstáculos poligonais convexos. Eles usam a função heurística para orientar a busca em direção ao objetivo, evitando obstáculos.

O desempenho depende da complexidade do ambiente, da qualidade da heurística e do algoritmo escolhido. Em cenários complexos com muitos obstáculos, a busca pode ser computacionalmente intensiva, mas ainda é eficaz para encontrar soluções ótimas ou quase ótimas. Além disso, a discretização do espaço contínuo em um grafo de visibilidade pode ser uma estratégia útil para simplificar o problema e melhorar o desempenho.

**3.9 a) Formule o problema precisamente, fazendo apenas as especificações necessárias para assegurar uma solução válida. Faça um diagrama do espaço de estados completo.**

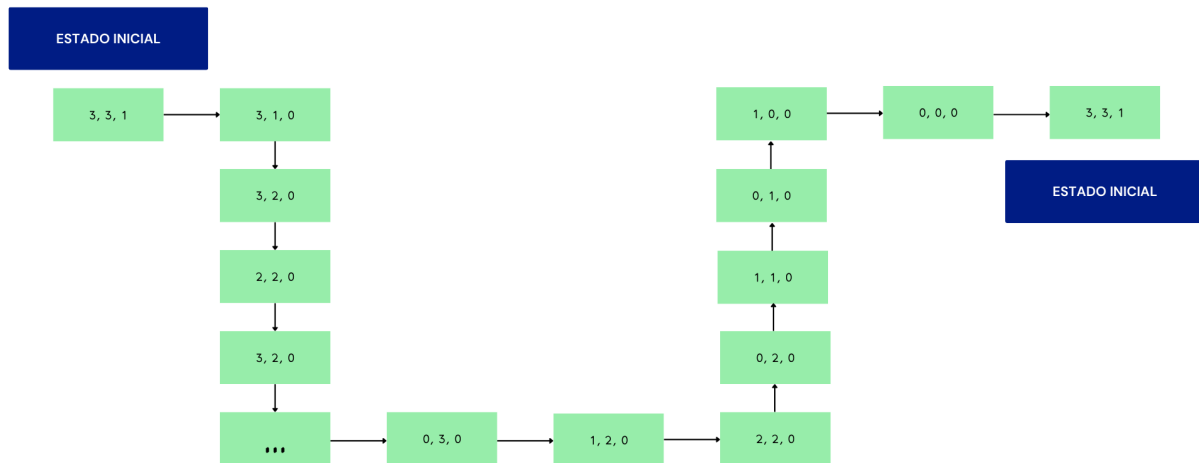
Estado Inicial: (3, 3, 1) - Representa três missionários, três canibais e um barco no lado inicial do rio.

Estado Final: (0, 0, 0) - Representa que todos os missionários, todos os canibais e o barco estão no lado oposto do rio.

Operadores: O barco pode carregar no máximo duas pessoas de qualquer combinação de missionários e canibais. No entanto, o número de canibais nunca deve exceder o número de missionários em nenhum lado do rio, caso contrário, os canibais comerão os missionários.

Restrições: Em qualquer lado do rio, o número de canibais não pode ser maior do que o número de missionários, a menos que não haja missionários nesse lado (ou seja, 0 missionários e 1 ou 2 canibais é aceitável).

Figura 1 - Diagrama



**b) Implemente e resolva o problema de forma ótima, utilizando um algoritmo de busca apropriado. É uma boa ideia verificar a existência de estados repetidos?**

Para resolver o problema de forma ótima, você pode usar um algoritmo de busca, como a busca em largura (BFS) ou a busca em profundidade (DFS), garantindo que estados repetidos sejam evitados. Como o espaço de estados é pequeno e não há custos associados às ações, você pode optar pelo BFS, que garante a menor quantidade de movimentos. A verificação de estados repetidos é importante para evitar loops infinitos. Se um estado já foi visitado antes, não há necessidade de explorá-lo novamente.

**c) Por que você imagina que as pessoas têm dificuldades para resolver esse quebra-cabeça, considerando que o espaço de estados é tão simples?**

Apesar do espaço de estados ser relativamente simples, as pessoas podem ter dificuldades em resolver esse quebra-cabeça por causa das restrições rígidas e do potencial para cometer erros que levariam a estados inválidos (por exemplo, ter mais canibais do que missionários em qualquer lado do rio).

**3.14 a) A busca em profundidade sempre expande pelo menos tantos nós quanto a busca A\* com uma heurística admissível.**

Falsa. A busca em profundidade não necessariamente expande tantos nós quanto a busca A\* com uma heurística admissível. A busca em profundidade expande os nós em profundidade antes de explorar outras opções, enquanto o A\* com uma heurística admissível pode direcionar a pesquisa para áreas mais promissoras, expandindo menos nós.

**b)  $h(n) = 0$  é uma heurística admissível para o quebra cabeças de 8 peças.**

Verdadeira.  $h(n) = 0$  é uma heurística admissível para o quebra-cabeças de 8 peças, pois nunca superestima o custo real para alcançar o objetivo. Isso ocorre porque a solução é sempre possível em um número finito de movimentos, e a heurística zero reflete essa situação.

**c) Em robótica, A\* não é útil porque as percepções, estados e ações são contínuas.**

Falsa. A\* é útil em robótica, mesmo quando as percepções, estados e ações são contínuas. O A\* pode ser adaptado para lidar com espaços de estados contínuos por meio da discretização do espaço ou do uso de métodos de otimização contínua. Ele ainda pode ser eficaz para planejamento de trajetória e resolução de problemas em ambientes contínuos.

**d) A busca em largura é completa mesmo se os custos de passos iguais a zero forem permitidos.**

Falsa. A busca em largura não é completa quando os custos de passos iguais a zero são permitidos, pois ela pode ficar presa em ciclos infinitos. Isso ocorre porque a busca em largura expande todos os nós no mesmo nível antes de passar para o próximo, e quando os custos são zero, ela pode entrar em loops infinitos.

**e) Assuma que a torre pode se mover em um tabuleiro de xadrez qualquer quantidade de quadrados em linha reta, verticalmente ou horizontalmente, mas não pode pular sobre as peças. A distância de Manhattan é uma heurística admissível para o problema de movimentar a torre do quadrado A para o B no menor número de movimentos.**

Verdadeira. A distância de Manhattan é uma heurística admissível para o problema de mover a torre em um tabuleiro de xadrez, pois nunca superestima o número mínimo de movimentos necessários para alcançar o objetivo. Essa heurística é eficaz para esse tipo de problema, onde a torre pode mover-se em linha reta, verticalmente ou horizontalmente, sem pular sobre peças, uma vez que reflete o número mínimo de quadrados que a torre deve percorrer para chegar ao destino.

**(b) Mostrar a derivada.**

$$\phi(v) = \frac{1}{1 + \exp(-av)}$$

$$\phi(v) = \frac{1}{1 + \exp(-av)}, \text{ mostrar que a derivada}$$

de  $\phi(v)$  em relação a  $v$  é dada por

$$\frac{d\phi}{dv} = a \phi(v) [1 - \phi(v)].$$

$$\frac{\partial \phi}{\partial v} = \frac{1}{1 + \exp(-av)}, \text{ aplicando propriedades}$$

dos expoentes:  $((1 + e^{-av})^{-1})$

aplicando a regra da cadeia:

$$\begin{aligned} \frac{-1}{(1 + e^{-av})^2} \cdot (-ae^{-av}) &= \frac{ae^{-av}}{(1 + e^{-av})^2} \\ &= \frac{1}{(1 + e^{-av})^2} (-ae^{-av}) \end{aligned}$$

simplificando:

$$\frac{-1}{(1 + e^{-av})^2} (-ae^{-av}) = \frac{ae^{-av}}{(1 + e^{-av})^2}$$